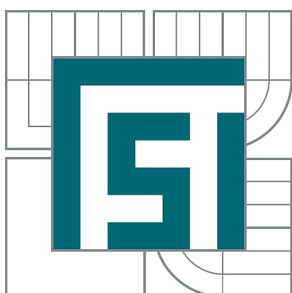




VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA STROJNÍHO INŽENÝRSTVÍ
ÚSTAV AUTOMATIZACE A INFORMATIKY

FACULTY OF MECHANICAL ENGINEERING
INSTITUTE OF AUTOMATION AND COMPUTER SCIENCE

NÁVRH A REALIZACE ŘÍZENÍ TRIBOMETRU

DESIGN AND REALIZATION OF TRIBOMETER CONTROL

BAKALÁŘSKÁ PRÁCE
BACHELOR'S THESIS

AUTOR PRÁCE
AUTHOR

LUKÁŠ JANÍK

VEDOUCÍ PRÁCE
SUPERVISOR

Ing. PAVEL HOUŠKA, Ph.D.

BRNO 2012

Vysoké učení technické v Brně, Fakulta strojního inženýrství

Ústav automatizace a informatiky

Akademický rok: 2011/2012

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

student(ka): Lukáš Janík

který/která studuje v **bakalářském studijním programu**

obor: **Aplikovaná informatika a řízení (3902R001)**

Ředitel ústavu Vám v souladu se zákonem č.111/1998 o vysokých školách a se Studijním a zkušebním řádem VUT v Brně určuje následující téma bakalářské práce:

Návrh a realizace řízení Tribometru

v anglickém jazyce:

Design and realization of Tribometer control

Stručná charakteristika problematiky úkolu:

Práce se zabývá návrhem a realizací software pro řízení Tribometru, realizovaného v laboratořích UK FSI VUT v Brně. Od realizovaného software se vyžaduje řízení a záznam průběhu experimentů, možnost jednoduchého plánování a ukládání plánu experimentu, včetně konfigurace Tribometru.

Cíle bakalářské práce:

1. Seznamte se s možnostmi ovládání řídicí jednotky motorů Unidrive pomocí sériové linky a ovládání realizujte
2. Seznamte se s možnostmi ovládání piezo-aktuátoru pomocí sériové linky a ovládání realizujte
3. Realizujte snímání otáček a řízení pohybu disku a kuličky Tribometru
4. Navrhněte strukturu řídicího software tribometru
5. Řídicí software realizujte a ověřte

Seznam odborné literatury:

- [1] Skalický J., Elektrické servopohony, Nakladatelství VUT, Brno
- [2] National Instruments: LabVIEW Control Design User Manual[online], 2010, 371057, dostupné z URL:<<http://www.ni.com>>
- [3] National Instruments: LabVIEW Fundamentals, August 2005, 374029C-01, dostupné z URL:<<http://www.ni.com>>

Vedoucí bakalářské práce: Ing. Pavel Houška, Ph.D.

Termín odevzdání bakalářské práce je stanoven časovým plánem akademického roku 2011/2012.

V Brně, dne 24.11.2010

L.S.

Ing. Jan Roupec, Ph.D.
Ředitel ústavu

prof. RNDr. Miroslav Doupovec, CSc., dr. h. c.
Děkan fakulty

ABSTRAKT

Bakalářská práce se zabývá návrhem a realizací softwaru pro tribologickou soustavu. S požadavkem vytvoření uživatelského prostředí pro zadání programu řízení pro daný experiment. Program je prováděn v určeném sledu a z experimentu jsou získány změřené data ze snímačů.

ABSTRACT

This Bachelor thesis deals with designing and realizing a software for tribology system. Requirement is to create user interface for programable control software for given experiment. Control software works in given order and measured data are obtained from experiment. The data are provided by sensors.

KLÍČOVÁ SLOVA

Labview, Tribologie, CompactRio, Piezo-aktuátor, Servomotor, Enkóder

KEYWORDS

Labview, Tribologie, CompactRio, Piezo-actuator, Servomotor, Encoder

PROHLÁŠENÍ O ORIGINALITĚ

Tímto prohlašuji, že předkládanou bakalářskou práci jsem vypracoval samostatně s využitím uvedené literatury a podkladů, na základě konzultací a vedení vedoucího práce

BIBLIOGRAFICKÁ CITACE

JANÍK, L. *Návrh a realizace řízení Tribometru*. Brno: Vysoké učení technické v Brně, Fakulta strojního inženýrství, 2012. 45 s. Vedoucí bakalářské práce Ing. Pavel Houška, Ph.D..

PODĚKOVÁNÍ

Děkuji vedoucímu bakalářské práce Ing. Pavlu Houškovi Ph.D. za pomoc a odborné vedení při vypracovávání této práce. Dále Ing. Martinu Zimmermanovi Ph.D. za odbornou pomoc při řešení požadavků na experiment a tvorbu uživatelského prostředí.

OBSAH

Abstrakt.....	5
Klíčová slova.....	5
Prohlášení o originalitě.....	7
Bibliografická citace	7
Poděkování.....	9
Obsah.....	11
1 Úvod.....	13
2 Tribologie.....	15
2.1 Kontaktní procesy.....	15
2.2 Procesy tření	15
2.3 Geometrie povrchu	15
2.4 Druhy tribologických zkoušek.....	16
3 Řízený tribometr	17
3.1 Konstrukce tribometru	17
3.2 Labview, Compact Rio	19
3.3 Řídicí jednotky Unidrive.....	19
3.3.1 Návrhy ovládání.....	19
3.4 Piezo-aktuátor	20
3.5 Snímače	21
3.5.1 Enkodéry.....	21
3.5.2 Snímač síly.....	21
3.5.3 Snímač vzdálenosti	21
3.5.4 Kamera.....	22
4 Řídicí software	23
4.1 Návrh řídicího software.....	23
4.2 Nastavení komunikace a rámců pro Unidrive SP.....	24
4.2.1 Realizace rámců pro jednotlivé parametry.....	26
4.2.2 Ovládání jednotek Unidrive SP se servomotory.....	28
4.3 Piezo-aktuátor	28
4.3.1 Tvorba trajektorií pro piezo-aktuátor	28
4.3.2 Inicializační příkazy.....	33
4.3.3 Rámce pro zápis a čtení z jednotky E-516.....	34
4.3.4 Logika programu pro řízení piezo-aktuátoru.....	34

4.4	Realizace řídicího softwaru	35
4.4.1	Uživatelské rozhraní.....	35
4.4.2	Nastavení	35
4.4.3	Manuální řízení	36
4.4.4	Nastavení trajektorie piezo-aktuátoru.....	37
4.4.5	Programové řízení.....	38
4.4.6	obsluha programu na cRiu	40
5	Závěr	43
6	Seznam použité literatury.....	44

1 Úvod

Tribologie je věda zabývající se vzájemným dotykem ploch a vlastnostmi mezi nimi. V našem případě se jedná o chování mazacího filmu mezi plochami. V současné době se využívá tribologie v dynamických dějích. Proto na Ústavu konstruování na FSI v Brně vznikl tento experiment tribologické soustavy. Požadavkem bylo umět řídit dané dynamické děje a umět měřit jejich, vlastnosti. Úkolem bylo vytvořit komplexní program, který by dokázal řídit několik různých zařízení soustavy a umět měřit dané veličiny, podle požadovaných nastavení a průběhů.

Pro daný problém jsme zvolili řešení od společnosti National Instrument, světové špičky v oblasti automatizace. Program jsme vytvářeli ve vývojovém prostředí Labview, což je moderní grafický programovací jazyk. Dále jsme využili reálnodobé řešení pomocí CompactRia s zásuvnými moduly pro komunikaci s různými druhy zařízeními a snímači. Toto řešení nám umožňuje velkou variabilitu při budoucích změnách v požadavcích na experiment a relativně snadné zakomponování nových zařízení a snímačů

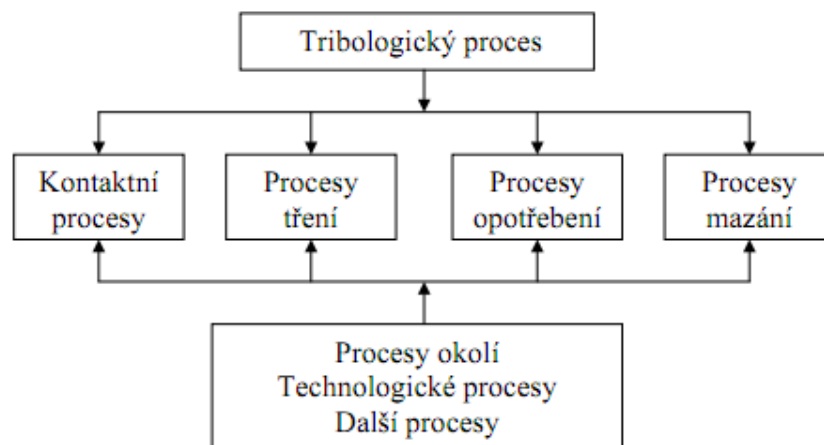
Tribologická soustava se skládá ze dvou os. Na každé ose je servomotor řízený frekvenčním měničem, na jedné ose je upevněná kulička a na druhé ose disk. Dále je zde piezo-aktuátor pro vytvoření přitlačné síly mezi diskem a kuličkou a pro tvorbu různých zatěžovacích trajektorií. Ke snímání povrchu mezi kuličkou a diskem je použita kamera. Za úkol je umět řídit všechna zařízení, pomoci uživatelem přednastaveného programu a umět zaznamenávat naměřená data.

Použité snímače na tribologické soustavě jsou Enkodéry na servomotorech, snímač síly a snímač vzdálenosti. Z daných snímačů realizuje zpracovávání dat, které se budou dále využívat, k další analýze.

2 Tribologie

Tribologie je vědní obor, který se zabývá chováním vzájemně dotýkajících se ploch a studuje jejich vlastnosti, jak ve statickém stavu, tak v dynamickém stavu. Kontaktní plochy mohou mít různé povrchy a také být vzájemně různých pohybech. Proces probíhá mezi třecími povrchy, mezi látkou a okolím.

Obecné vazby mezi jednotlivými částmi tribologických procesů jsou znázorněny na obr. 1. [1]



Obr. 1 Vzájemné vazby v tribologickém systému.[1]

2.1 Kontaktní procesy

Důležitým faktorem je velikost stykové plochy, ty jsou závislé na vzájemném styku těles tribologické soustavy. Kde musíme uvažovat několik základních vlivů, jako jsou fyzikální, chemické a mechanické vlastnosti. Dále počet těles soustavy a jejich různé druhy deformací mezi jednotlivými tělesy. Dále různé typy vzájemných relativních pohybů. [1]

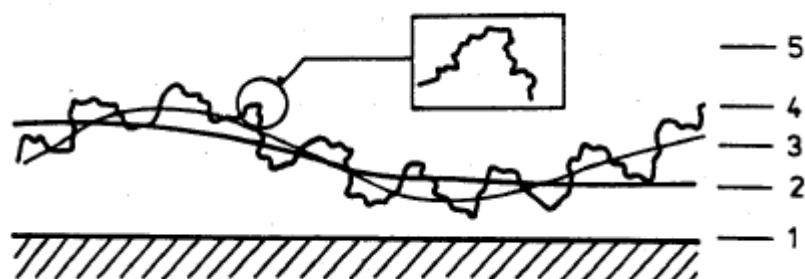
2.2 Procesy tření

Tření se váže k vzájemnému relativnímu pohybu dotýkajících se těles tribologického systému. Tento jev můžeme zkoumat na úrovni mikromodelu, kde sledujeme elementární proces, nebo na úrovni makromodelu, kde sledujeme celý funkční děj.

Máme čtyři základní stavy tření, tření tuhých těles (suché tření), tření kapalinové, tření plynné a tření plazmatické. V praxi většinou se vyskytují kombinace těchto druhů tření.[1]

2.3 Geometrie povrchu

Kvalitu povrchu vyjadřujeme jako geometrické, fyzikální, chemické a další vlastnosti. Na obr. 2 vidíme znázorněny geometrie povrchu.



Obr. 2 Znázornění geometrie povrchu

1- dokonale rovný povrch, 2- odchylky tvaru a polohy, 3- vlnitost povrchů, 4- mikroskopické drsnosti, 5- submikroskopické nerovnosti [1]

Drsnost povrchu je jeden ze základních způsobů hodnocení povrchu sledované plochy a je většinou charakterizovaný normou. [1]

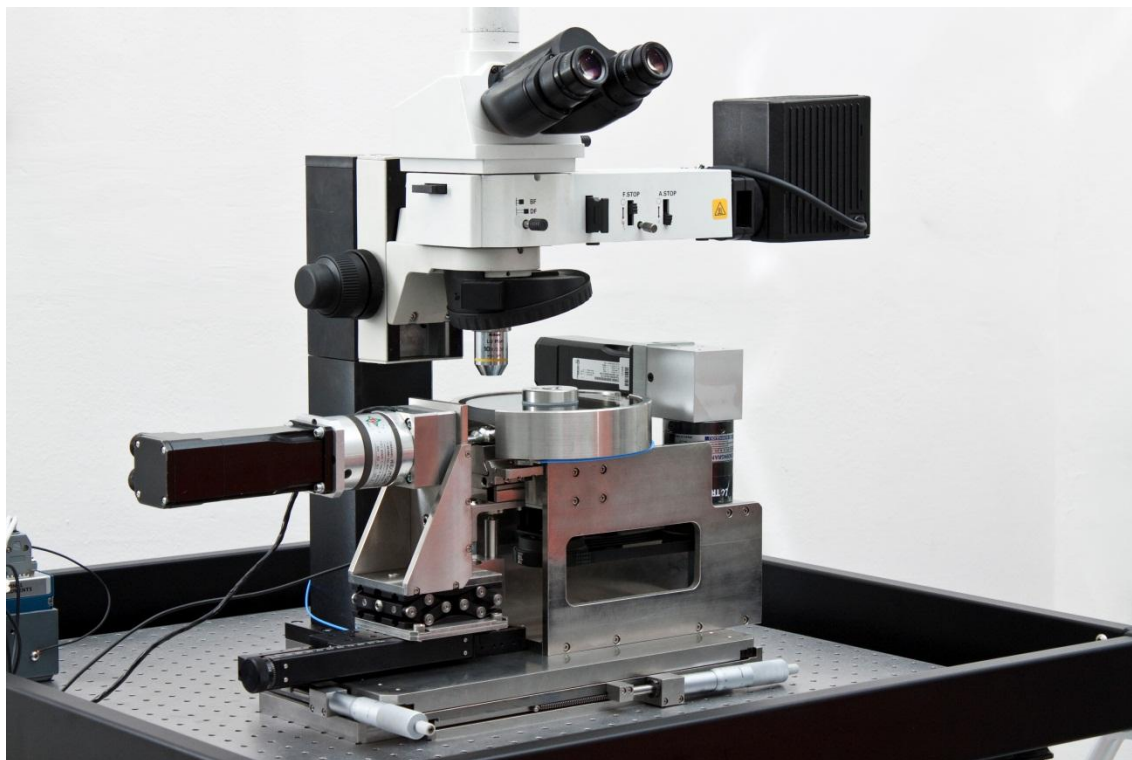
2.4 Druhy tribologických zkoušek

Máme k dispozici mnoho druhů tribologických zkoušek, proto volíme vhodnou pro daný experiment a snažíme se získat co nejpřesnější metodu pro reálné soustavy. Proto získané údaje na jednoduchých tribometrických měření se nemusí shodovat s reálnou soustavou, protože neznáme všechny faktory.[1]

3 Řízený tribometr

3.1 Konstrukce tribometru

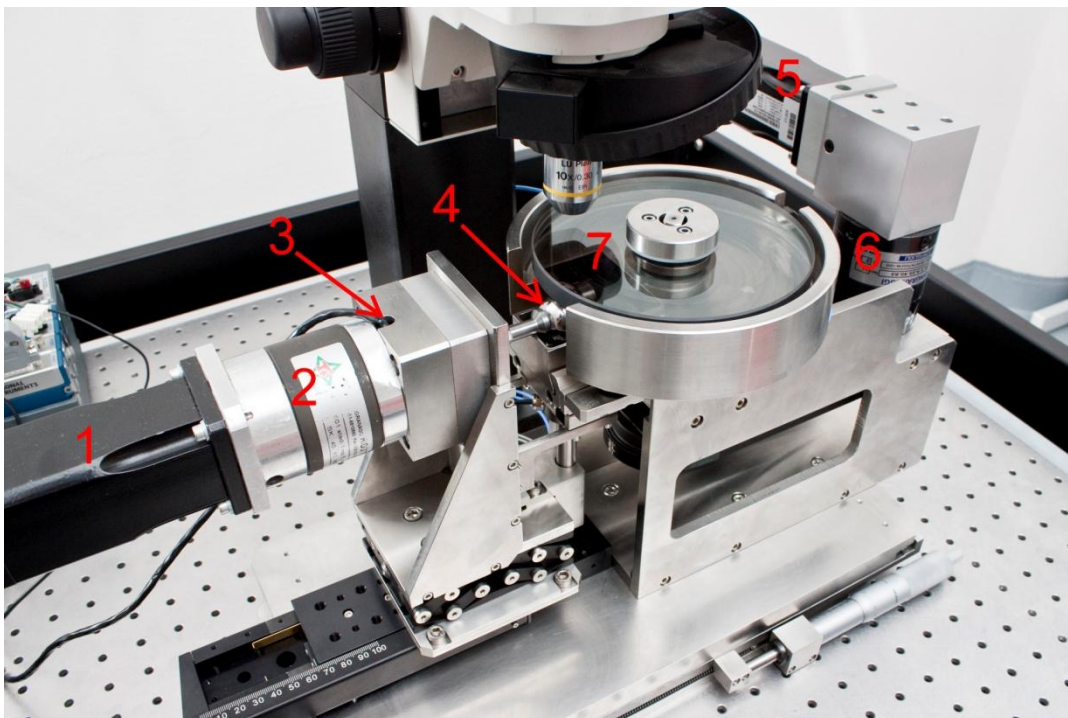
Tribometr je vyfocený na obr. 3, kde je vidět daná tribometrická soustava. Konstrukce tribometru byla navržena na Ústavu konstruování na FSI v Brně.



Obr. 3 tribometr

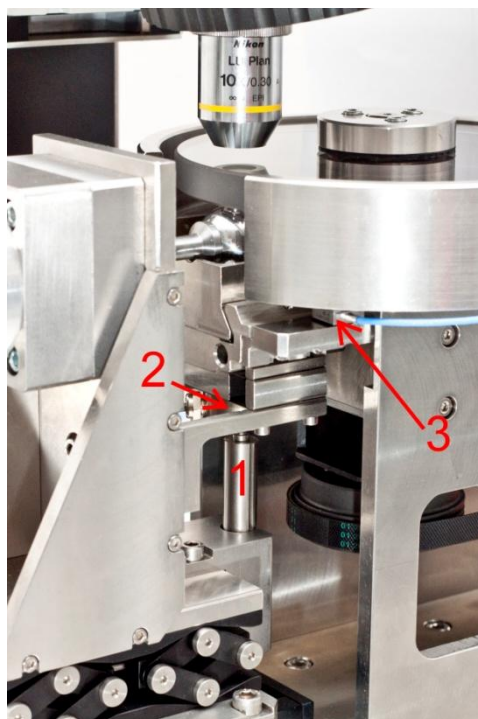
Na obr. 4 vidíme jednotlivé pohony os. Pod číslem 1 je servomotor pro pohon kuličky. Kulička je vidět pod číslem 4. Číslo 2 je převodovka ke kuličce, která má převodový poměr 5:1. 3 je enkodér pro kuličku, který není vidět, protože je uvnitř pouzdra držáku pro motor. Celá soustava pohonu kuličky je ve vodorovné poloze uchycena ve stojanu. Stojan je pohyblivý, kde můžeme nastavovat výšku, která je limitována stykem kuličky s diskem. Další pohyb je posun stojanu po horizontální dráze, pro vysunutí kuličky od disku. Kulička je ve vaničce, do které se umísťuje měřené mazivo.

Pod číslem 5 je vidět část servomotoru pro pohyb disku. 6 je převodovka pro disk, která má převodní poměr 20:1. Skleněný disk je pod číslem 6. Enkodér disku není prozatím nainstalován, kvůli chybě konstrukce, která bude odstraněna. Na převodovce disku je řemenice, která je pomocí řemene spojena s řemenicí disku. Kamera se umísťuje do tubusu nad mikroskop, tubus je vidět na obr. 3.



Obr. 4 pohon kuličky, disku

Na obr. 5 je vidět umístění piezo-aktuátoru pod číslem 1. proti němu je umístěný snímač síly, číslo 2. Nad snímačem síly je umístěna lamelová pružina a na ní je vanička pro kuličku. Snímač síly měří jak sílu od piezo-aktuátoru, tak od přtlaku kuličky k disku. Pod číslem 3 je umístěn snímač polohy, který měří vzdálenost mezi stojanem kuličky a částí konstrukce disku. Snímač má měřit vzdálenost mezi diskem a kuličkou.



Obr. 5 piezo-aktuátor, snímač síly, snímač vzdálenosti

3.2 Labview, Compact Rio

Pro realizaci základního ovládání jako pro celkovou aplikaci jsme, použili software od společnosti National Instrument. Software se jmenuje Labview a je to objektový programovací jazyk. Dále máme použít embedded systém, také od společnosti National Instrument. Embedded systém je CompactRio 9024 (cRio) s šasi pro čtyři zásuvné moduly. Šasi máme plně obsazeno zásuvnými moduly. Moduly jsou NI 9870, NI 9239, NI 9237, NI 9411.

Při práci v Labview vytváříme programy zvané VI (Virtual Instrument), pod programy jsou nazývány subVI. [2][3][4][5][6]

3.3 Řídicí jednotky Unidrive

Obě řízené osy jsou osazeny elektrickými pohony M256D (obr. 7) s řídicími jednotkami Unidrive SP1201 (obr. 6) fa. Control Techniques. Jednotky jsou napájeny ze sítě 230V a jsou osazeny komunikačními moduly s rozhraním RS232. Servomotory jsou připojeny přes napájecí kabel a datový kabel. Řízení můžeme provádět přes displej, který je přímo na každém měniči a podle zvolených parametrů jej programovat nebo přímo ovládat. Tento druh řízení není moc vhodný, protože můžeme nastavovat jen jeden parametr i když je možné si přednastavit program do měniče a dále si ho spustit. Nevýhodou je, že stále můžeme sledovat jen jeden parametr např. rychlost. Proto jsme využili připojení pomocí sériové linky s počítačem a naprogramovali komunikační rámec pro základní ovládání, což je popsáno níže. [7][8]



Obr. 6 Unidrive SP1201[9]

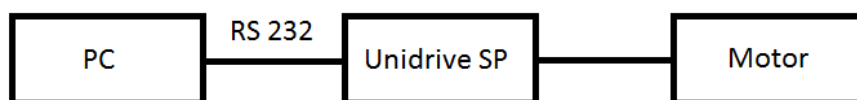


Obrázek 7 Servomotor M256D[10]

3.3.1 Návrhy ovládání

Jednotku Unidrive SP jsme připojili pomocí sériové linky přímo s počítačem. Komunikace byla realizována přes sériový port v počítači. Komunikace byla nejprve navržena pomocí protokolu VISA, který byl využit pro komunikaci mezi počítačem a

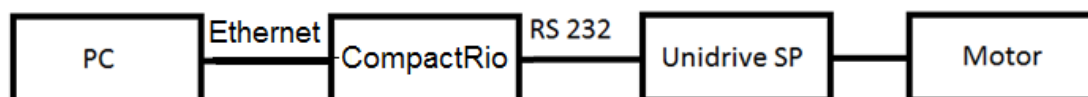
jednotkou Unidrive SP (obr. 8). Toto zapojení jsme používali k seznámení a odzkoušení zápisu a vyčítání parametrů pro ovládání měniče. [7]



Obr. 8 Blokové schéma zapojení Unidrive SP

Pro konečné řešení jsme využili propojení přes cRio od National Instrument. Komunikaci s jednotkami Unidrive je udělaná zásuvným modulem NI 9870. Tento modul je 4 kanálový sériový port RS232. Kde na prvním kanálu je připojena jednotka s kuličkou a druhým kanálu jednotka s diskem. [2] [3]

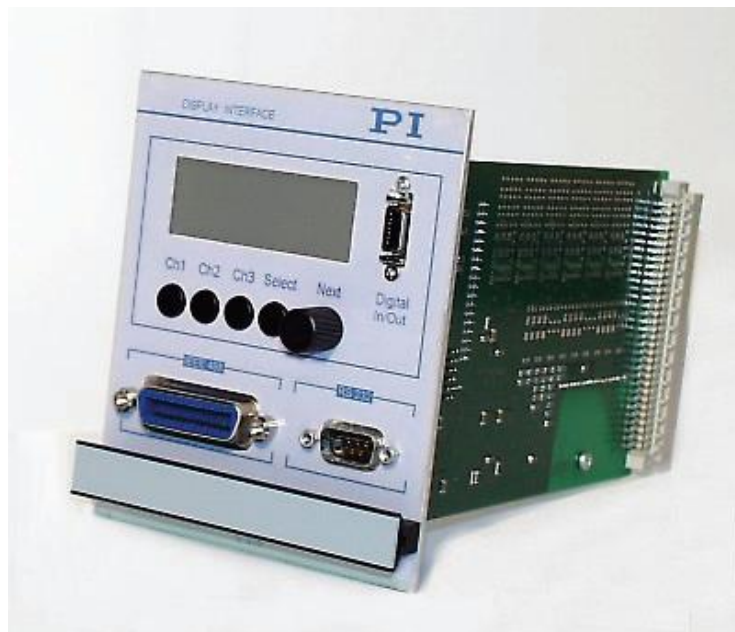
Komunikační protokol je využit UART, který je naprogramovaný na FPGA cRia. Blokové schéma komunikace je znázorněno na obrázku 9.



Obrázek 9 Blokové schéma zapojení Unidrive SP přes CompactRio

3.4 Piezo-aktuátor

Piezo-actuator je od firmy Physic-Instrument. Hlavní jednotka pro řízení se jmenuje Computer Interface and Display Module je to model E-516 (Obr. 10). Tento model už se nevyrábí je nahrazen modelem E-517. Modul je vybaven 6 místním displejem, který umožňuje zobrazovat určitá základní data. Dále komunikačními porty IEEE488 a RS-232. Tato jednotka je dále osazena dalšími moduly podle požadavků aplikace nebo zákazníka. V našem případě je osazen zesilovacím modulem E-503, který je jednokanáloví. Další modul je Servo-Controller E-509, také jednokanáloví. [11]



Obr. 10 jednotka E-516 [11]

Jednotku máme připojenou přes konektor RS-232 s počítačem. Tlačítka pod displejem na obr. 10, používám k přepínání jednotlivých kanálů, vstupů piezo-aktuátorů v našem případě máme jen jeden kanál. Otočným přepínačem na straně měníme funkce daného kanálu, tyto funkce jsou závislé na zvoleném operačním režimu. Máme dva operační režimy Offline a Online. V offline režimu, ovládáme piezo-aktuátor manuálně pomocí tlačítka DC-Offset, které nastavuje napětí na výstup a tím ovládá aktuátor. V online režimu nastavujeme ovládání piezo-aktuátoru vnějšími příkazy, které můžeme posílat přímo z počítače, kde máme nainstalovaný příslušný software. [11]

Dále můžeme využít Servo režim, ten jde spouštět jenom v online režimu. V servo režimu ON, ovládáme vstup pomocí hodnot pozic aktuátoru. V servo režimu OFF ovládáme vstup piezo pomocí úrovně napětí. [11]

3.5 Snímače

3.5.1 Enkodéry

Snímání otáček je řešeno pomocí inkrementálních enkodérů. Enkodéry jsou připojené přes měřicí zásuvný modul NI 9411. Tato měřicí karta má 6 vstupů a vzorkovací rychlost je 500ns. Encodery jsou připojeny na vstupech 1 a 2. [6]

Z enkodérů se inkrementálně vyčítají vzorky, s kterými dále pracujeme. Směr otáčení servomotorů nám zajišťuje naprogramovaná logika.

3.5.2 Snímač síly

Pro snímání síly je použit tenzometr, který je připojen přes zásuvný modul NI 9237. Karta umožňuje zapojení jako half-bridge a full-bridge. V našem případě je senzor zapojen jako full-bridge. NI 9237 nám měří napětí z tenzometru, které přepočítáváme na sílu [5]

3.5.3 Snímač vzdálenosti

Jedná se o kapacitní senzor vzdálenosti od společnosti Micro-Epsilon. Snímač je připojen přes zásuvný modul NI 9239, která má analogový vstup. Vstupní rozsah je $\pm 10V$,

rychlost je 50000 vzorku za sekundu na vstup. Snímání je realizováno pomocí napěťové smyčky. [4]

3.5.4 Kamera

Kamera běží nezávisle v samostatném programu. Ale máme možnost vnějšího spouštění, které jsme využili. K tomu využíváme zásuvný modul NI 9870, kde na čtvrtém kanálu je připojena ke kameře. Pro spouštění kamery generujeme požadovaný puls na spuštění. [3]

4 Řídící software

4.1 Návrh řídicího software

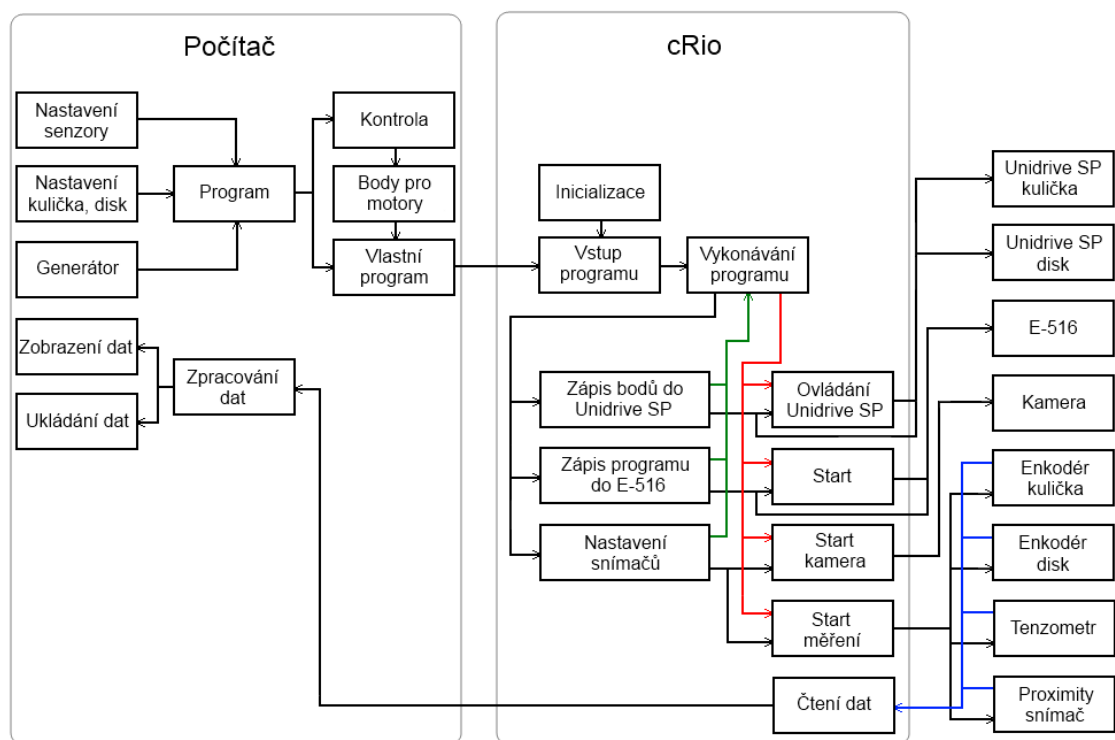
Řídící software byl navrhován podle daných požadavků a případné variability pro budoucí úpravy. Cíli software je navržený v programu Labview od společnosti National Instrument. Na počítači má být realizováno ovládání programu. CRio bude obsluhovat jednotlivá zařízení získávat jednotlivá data ze senzorů.

Na obr. 11 je znázorněno blokové schéma ovládání programu. V části počítače vidíme nastavení a tvoření vlastního programu. V bloku program si vybíráme z dvou možných řízení a to programového a manuálního. Jestliže zvolíme programové, vytvoříme si daný program podle požadavků, k tomu musíme nastavit, nastavení senzorů a nastavení kuličky s diskem. Generátor je pro tvoření trajektorií piezo-aktuátoru, který podle požadavků můžeme využít v programu. Po vytvoření programu se program zkontroluje a vygenerují se jednotlivé body pro motory a ve vlastním programu jednotlivé příkazy pro daná zařízení. V bloku vlastního programu se přiřadí nastavení a program pro trajektorii piezo-aktuátoru. Celý vytvořený program s nastavením se pošle na cRio.

CRio se nejdříve nastaví do výchozího stavu a provede inicializaci komunikace s jednotlivými zařízeními. Po té se čeká na příchozí program od uživatele. Po příchodu programu se zjišťuje, o jaký druh jde, v našem případě máme dva druhy, programový a manuální, který jsme už zvolili v uživatelském rozhraní na počítači. Po té se začne daný program vykonávat, v prvním kroku se zapíše jednotlivé nastavení a příkazy a body programu do jednotlivých zařízení, program čeká do doby, než se všechny příkazy provedou, v schématu je to znázorněno zelenou barvou. Až se vrátí odpověď od všech zařízení, že skončili daný proces zápisu, tak se začne vykonávat vlastní program.

Vlastní program se vykonává v posloupnosti příkazů, které se posílají v daném sledu jednotlivým zařízením, do doby než jsou provedeny všechny příkazy a program na cRio je ukončen. Jestliže bylo zapnuto měření, tak se vyčtou jednotlivá data, která se následně posílají do počítače, kde jsou dále zpracována.

Získané data z měření se v počítači dále zpracovávají, jednotlivá data se přepočítají na požadované výstupní hodnoty. Ty se nám zobrazí v grafu měření, jestli byl při spuštění programu nastaven zápis do souboru, tak se nám jednotlivá data uloží v textovém souboru.



Obrázek 11 blokové schéma programu

4.2 Nastavení komunikace a rámců pro Unidrive SP

Pro komunikaci používáme protokol VISA od Společnosti National Instrument. Jednotka Unidrive SP podporuje typ komunikačního protokolu ANSIx3.28, jež umožňuje přístup k 32bitovým parametrům. V tab. 1 máme zobrazenou vrstvu UART a její nastavení. [7]

Atribut	Popis
Fyzická vrstva	4 vodičové RS 232
Bitový tok	Standardní UART (Universal Asynchronous Receiver Transmitter) asynchronní znaky s NRZ (Non Return to Zero)
Znak	Každý znak se skládá z: 1 start bit 7 datových bitů (ASCII) 1 paritní bit (sudá parita) 1 stop bit
Poměr Baudů	300, 600, 1200, 2400, 4800, 9600, 19200, 38400

Tab. 1 Fyzická vrstva a UART [7]

Čtení je na principu dotazu na daný parametr, který vrácí svojí hodnotu. Všechny parametry a jejich vlastnosti jsou podrobně popsány v příručce od měničů Unidravive SP. Dotaz je posílán ve tvaru, který je znázorněn v tab. 2. [7]

EOT	Konec přenosu (End of transmission) (Ctl D)
A1	Adresa měniče: 1. číslice
A1	Adresa měniče: 1. číslice
A2	Adresa měniče: 2. číslice
A2	Adresa měniče: 2. číslice
M1	Číslo menu: 1. číslice
M2	Číslo menu: 2. číslice
P1	Číslo parametru: 1. číslice
P2	Číslo parametru: 2. číslice
ENQ	Dotaz (Enquiry) (Ctl E)

Tab. 2 Příkaz pro čtení parametru [7]

Odpověď je posílána ve tvaru, která je znázorněna v tab. 3. Délka datového typu je proměnlivá, ale maximální hodnot nesmí překročit 12 znaků, v kterém jsou zahrnuty znaménka a desetinné čárky. Odpověď není vrácena, když je chybná adresa měniče. Když se odkazujeme na neexistující menu nebo parametr, pak je vrácena hodnota EOT, což je konec přenosu. Kontrolní součet je tvořen 8 bitovou hodnotou, pokud je hodnota menší než 32, tak se k ní připočte 32. [2]

Kontrolní součet: $M1 \wedge M2 \wedge P1 \wedge P2 \wedge \dots \wedge Dn \wedge ETX$ [2]

STX	Začátek textu (Start of text) (Ctl B)
M1	Číslo menu: 1. číslice
M2	Číslo menu: 2. číslice
P1	Číslo parametru: 1. číslice
P2	Číslo parametru: 2. číslice
D1	Data: 1. číslice
D2	Data: 2. číslice
-	
-	
Dn	Data: n-tá číslice
ETX	Konec textu (End of text) (Ctl C)
	Kontrolní součet

Tab. 3 Tvar odpovědi [7]

Příkaz pro změnu parametru je znázorněn v tab. 4. Zápis hodnoty parametru se řídí několika pravidly. Délka nesmí být víc jak 12 znaků, pole musí mít úvodní mezery. Znaménkový znak a čárka nepovinná. Velikost nesmí přesáhnout 2^{-31} až $2^{31}-1$. Je-li hodnota správně zapsána, vrací se znak potvrzení ACK, jestliže nejsou splněni výše uvedené podmínky tak se vrací znak NAK. Kontrolní součet je dělán stejně jako u čtení a řídí se stejnými pravidly. [2]

EOT	Konec přenosu (End of transmission) (Ctl D)
A1	Adresa měniče: 1. číslice
A1	Adresa měniče: 1. číslice
A2	Adresa měniče: 2. číslice
A2	Adresa měniče: 2. číslice
STX	Začátek textu (Start of text) (Ctl B)
M1	Číslo menu: 1. číslice
M2	Číslo menu: 2. číslice
P1	Číslo parametru: 1. číslice
P2	Číslo parametru: 2. číslice
D1	Data: 1. číslice
D2	Data: 2. číslice
-	
-	
Dn	Data: n-tá číslice
ETX	Konec textu (End of text) (Ctl C)
	Kontrolní součet

Tab. 4 Zápis parametru [7]

4.2.1 Realizace rámců pro jednotlivé parametry

Pro zápis požadovaných parametrů jsme vytvořili SubVI, které posílá dané rámce dat. SubVI má vstupy: menu, parametr, požadovaná hodnota a formát zadávané hodnoty. Ze vstupů vytvoří požadovaný rámec pro danou jednotku Unidrive.

Všechny parametry musejí být zapsány ve správném formátu, to znamená, že musí být zadány správné desetinné místo. Nevyplněné desetinné místa se doplňují nulami na požadovaný formát.

4.2.1.1 Nastavení řídicího slova

Před zahájení posílání a vyčítání jednotlivých příkazů, nastavujeme měnič do stavu povoleno ovládání pomocí řídicího slova. Toto nastavení je v menu 6 a parametru 43, hodnota se nastavuje na 1 (dále menu a parametr bude zapisován jako 6.43, kde první číslo je menu a druhé je parametr). [7]

4.2.1.2 Nastavení PID regulátoru

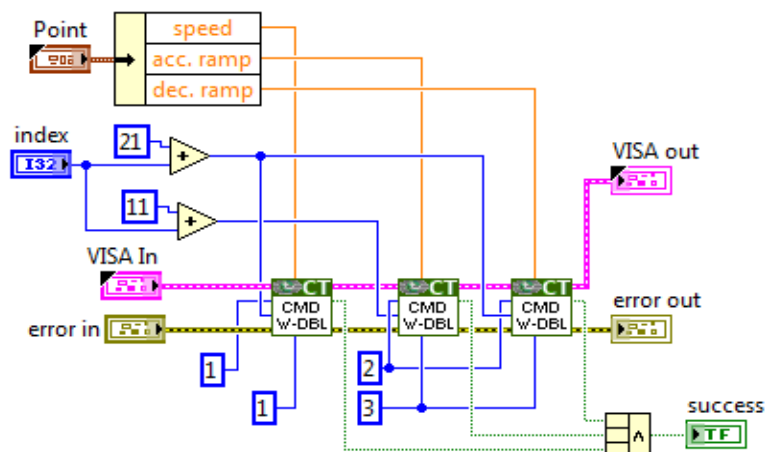
Důležitým krokem je nastavení PID regulátoru pro vlastní řízení motorů. Využili jsme, k nastavení metodu Ziegler-Nichols.

Menu pro zadávání PID parametrů je 3.10 pro proporcionální složku, 3.11 pro integrační složku a 3.12 pro derivační složku. Zde jsme byli omezeni rozsahem zadávaných hodnot jednotlivých parametrů, složka P a D má rozsah od 0-6,6335. Integrační složka od 0-653,35. Regulátor je nastaven: P=0,011 I=1,4 D=0,004. [7]

4.2.1.3 Zápis jednotlivých rychlostí a zrychlení

Použité měniče Unidrive SP1201 mají 8 míst pro přednastavené otáčky, 8 míst pro akcelerační a decelerační rampy. Otáčky jsou omezeny na hodnotu od -3300 do 3300ot/min. Rampy mají omezení v rozsahu od 0 do 3200,000 s/1000 ot/min [7]

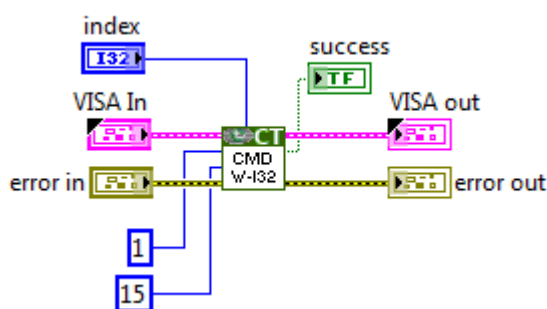
Přednastavené otáčky zapisujeme do parametrů 1.21-1.28 do každé pozice můžeme zapsat pouze jedinou hodnotu otáček. Parametry pro akcelerační rampy jsou 2.11-2.18 a pro decelerační rampy 2.21-2.28. Na obr. 7 je vidět subVI, které obsluhuje zápis jednotlivých rychlostí a ramp do měniče. Tyto body zapisujeme po blocích, kde je zapsána rychlost i akcelerační a decelerační rampa dohromady, které přicházejí ze vstupu point. Vstup index udává, do jakého parametru se mají body zapsat. [7]



Obrázek 12 zápis rychlostí a ramp

4.2.1.4 Výběr jednotlivých rychlostí a zrychlení

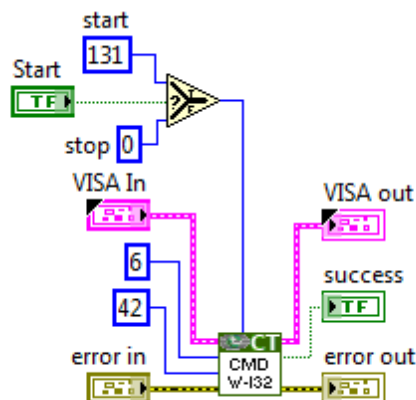
Volba přednastavených otáček je v menu 1.15. pro výběr správné rampy musíme zvolit příslušný index parametru, například pro výběr první rychlosti z paměti použijeme index 1, který se odkazuje na zapsanou rychlost v menu 1.21. Poslední index přednastavených otáček je 8. Výběr akceleračních a deceleračních ramp je na stejném principu jako přednastavené otáčky. Menu pro výběr akcelerační rampy je 2.10 a decelerační rampy 2.20. Na obr. 13 je subVI pro výběr dané rychlosti, kde požadovanou rychlost vybíráme pomocí hodnoty index. [7]



Obr. 13 výběr přednastavené rychlosti

4.2.1.5 Parametry start a stop

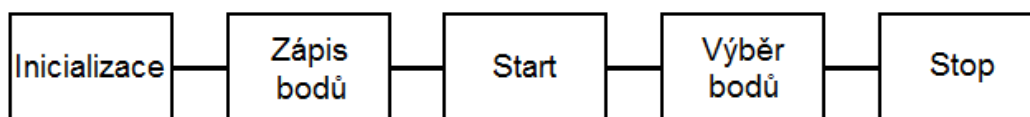
Parametr start nám umožňuje odstartovat motory a poté už můžeme jen vybrat zvolené přednastavené otáčky, tím máme zajištěnou kontinuální běh motorů. Po skončení požadovaných průběhu zastavíme motory Příkazem stop. Na obr. 14 vidíme subVI pro obsluhu příkazu start a stop. Tento příkaz je v měniči na adrese 6.42 a hodnota pro start je 131, pro stop 0. [7]



Obrázek 14 start, stop

4.2.2 Ovládání jednotek Unidrive SP se servomotory

Logika ovládání je vidět na obr. 15, kde je vidět blokové schéma řízení servomotorů. Prvním krokem je inicializace, kde se nastaví komunikace a základní nastavení parametrů měniče. Zde se nastaví řídicí slovo a také nastaví PID regulátor. V dalším kroku se zapíše všechny rychlosti a rampy do požadovaného měniče, kde jsou uloženy pod požadovaným indexem. V kroku Start, se spustí motory a ty čekají na jednotlivé indexy rychlostí a ramp, které se vyberou v dalším kroku. Podle požadavku programu vybereme požadované body v určených časových intervalech. Po skončení programu se motory zastaví pomocí stop a vyčká se na opětovné spuštění, při dalším spuštění už se inicializace znovu neprovádí. Body se také nezapisují, jestliže zůstali stejné a tedy nebyla provedena úprava v programu.



Obr 15 blokové schéma řízení servomotorů

4.3 Piezo-aktuátor

Pro základní seznámení jsme využili propojení počítače a E-516 přes sériovou linku. Pro základní ovládání jsme využili programovou sadu, která byla doručena s jednotkou. Zde jsme měli několik možností z výběru softwaru, pro ovládání. Jelikož naši aplikaci děláme v Labview, tak jsme zvolili tuto možnost a využili jsme VI, která byla dodána. První VI bylo konfigurační, které nastavovalo jednotku E-516 do ovládání pomocí počítače. Dále jsme měli možnost si vybrat z několika ukázkových aplikací. Pro naše účely jsme nejdříve spustili PI Terminal, v kterém jsme si mohli odzkoušet posílání základních příkazů. Další důležité VI bylo WaveGenerator, v kterém jsme mohli nastavit různé průběhy pro piezo-aktuátor. [11]

4.3.1 Tvorba trajektorií pro piezo-aktuátor

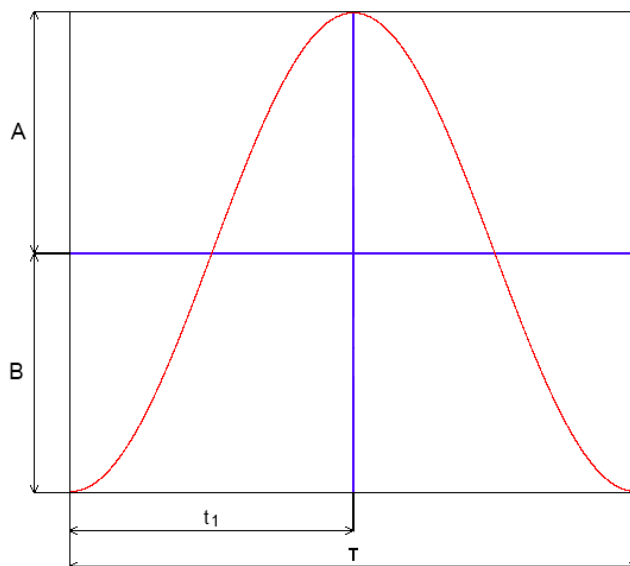
4.3.1.1 Sinusový trajektorie piezo-aktuátoru

Pro tvoření sinusového trajektorie používáme popsanou matematiku, která byla přiložena v manuálu. Na obr. 16 vidíme sinusovou trajektorii, kde A je amplituda v mikrometrech, B je offset v mikrometrech, t_1 čas náběžné hrany, T je perioda. Dalším

důležitou hodnotou je počet bodů na periodu. Maximální počet bodů na periodu je 8192 To získáme ze vzorce: [12]

$$P = T/dT \quad (1)$$

Kde T je čas periody a dT je minimální časová změna jednotky E-516. P je počet bodů pro danou periodu. [12]



Obr. 16 sinusová funkce

Základní příkaz pro konstrukci sinusové trajektorie je:

$$SIN = A, N_p, x_0, \emptyset, B$$

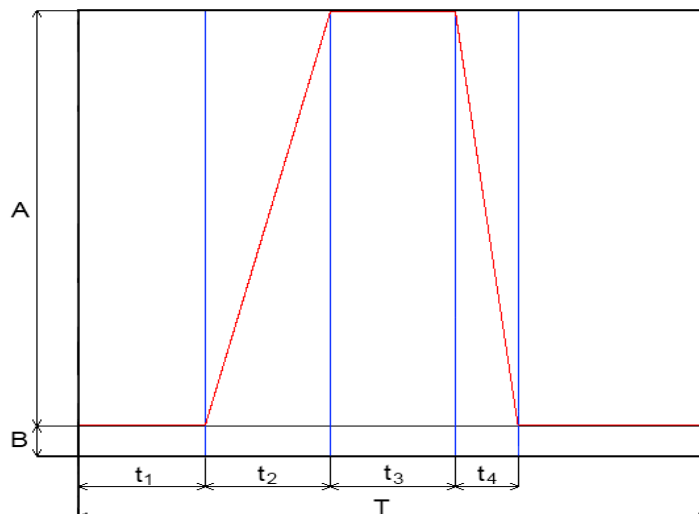
A amplituda v μm , N_p je délka průběhu v bodech, x_0 je počáteční bod, \emptyset úhel v stupních, B offset v μm . [11]

WAV A SIN $x_1, x_2, x_3, x_4, x_5, x_6, x_7$
WAV A SIN 0 6410 10 6410 0 0 20

Příkaz ukázky sinusové trajektorie, která se posílá do jednotky E-516. Příkaz začíná slovem WAV, což je řídicí příkaz pro generátor za ním následuje písmeno A, které udává požadovanou pro E-516. SIN vyjadřuje, že se jedná sinusovou trajektorii. První parametr je x_1 (0) což udává počáteční bod trajektorie, další parametr je x_2 (6410), který udává celkový počet bodů na periodu. Parametr x_3 (10) je amplituda A . Další parametr x_4 (6410) je počet bodů sinusové trajektorie. Další dva parametry x_5, x_6 (0, 0) jsou počáteční bod a úhel ve stupních. Posledním parametrem x_7 je offset. [11]

4.3.1.2 Lichoběžníková trajektorie piezo-aktuátoru

Z lichoběžníkové trajektorie, můžeme vytvořit i další průběhy jako jsou různé druhy skokových nebo pilových trajektorií. Lichoběžníkový trajektorie je na obr. 17, na kterém si popíšeme výpočet a vytvoření příkazů. [11] [12]



Obr. 17 lichoběžníková trajektorie

T je perioda trajektorie, A je amplituda v μm , B je offset v μm , t_1 je čas zpoždění v procentech, t_2 je čas náběžné hrany v procentech, t_3 je čas horní hrany v procentech, t_4 je čas sestupné hrany v procentech. Podmínkou je: $T \leq t_1 + t_2 + t_3 + t_4$ [11]

Lichoběžník se skládá z polynomickeho příkazu, který má tvar:

$$POL = x_0, A_1, A_2, A_3, A_4, A_5$$

x_0 je počáteční bod, A_1 až A_5 jsou volitelné hodnoty podle požadované funkce. [12]

4.3.1.3 Výpočet periody lichoběžníkového trajektorie

Důležitou hodnotou pro nás je celková doba periody T , z které si musíme vypočítat celkový počet bodů pro danou trajektorii. Dále musíme přiřadit, ke každé hraně a zpoždění určitý počet bodů podle požadované trajektorie. Na obr. 17 vidíme jednotlivé časy t_1 až t_4 , které nám určují dobu části trajektorie. Časy jsou zadávány v procentech proti původní periodě.

Příklad: Máme puls z obr. 14, kde, $T=1\text{s}$, $t_1=20\%$, $t_2=20\%$, $t_3=20\%$, $t_4=10\%$, amplituda=20, offset=20.

Nejdříve si vypočítáme P :

$$P = \frac{1}{5,2 * 10^{-5}} = 19230,8 \quad (2)$$

Jenomže maximální hodnota P může být jen 8192. Takže pro tento případ musím použít, další vztah, kterým získáme konstantu na dělení. Tato operace se provádí pro jakoukoliv trajektorii, která přesáhne maximální hodnotu bodů periody. [11] [12]

$$K \cong \frac{P}{P_{max}} ; \frac{19230,8}{8192} = 2,35 \cong 3 \quad (3)$$

K je konstanta pro dělení, P je hodnota z předchozího výpočtu, P_{max} je maximální hodnota bodů na periodu (8192). Hodnota K se vždy zaokrouhluje nahoru a na celé číslo.

$$P_k = \frac{P}{K}; P = \frac{19230,8}{3} = 6410,27 \cong 6410 \quad (4)$$

P_k je počet bodů na periodu, P je celkový počet bodů a K je konstant pro dělení.

Nyní máme požadovaný počet bodů pro trajektorii. Dále si spočítáme počty bodů, pro jednotlivé části trajektorie. Pro to použijeme jednoduchý přepočet. [11] [12]

$$P_x \cong P_k * t_x; P_1 \cong 6410 * 0,2 \cong 1282 \quad (5)$$

P_x je požadovaná hodnota bodů pro danou část trajektorie, t_x je délka průběhu v desetinách. P_k je počet bodů na periodu. Výsledná hodnota musí být celé číslo. [11] [12]

4.3.1.4 Tvorba příkazů pro lichoběžníkové trajektorie

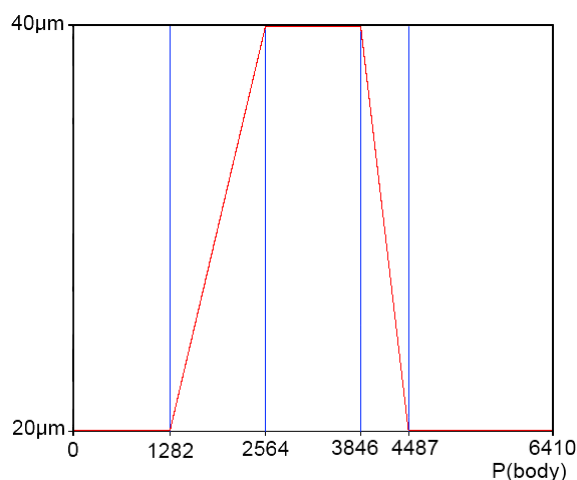
Jak už bylo zmíněno lichoběžníková trajektorie, je tvořena z polynomů. Pro náš příklad vypadají příkazy následovně. [11]

```
WAV A POL 0 1282 0 20.0000
WAV A POL 1282 1282 1282 20.0000 0.156006
WAV A POL 2564 1282 0 40.0000
WAV A POL 3846 641 3846 40.0000 -0.0312012
WAV A POL 4487 1923 0 20.0000
```

Příkazy mají totožnou strukturu jednotlivých členů, vždy na prvním místě je počáteční bod daného kroku, další člen je počet bodů první části trajektorie. Třetím členem je počet bodů náběžné nebo sestupné hrany, v ostatních případech je tento parametr 0. Člen 4 je pozice (vysunutí) piezo-aktuátoru na začátku dané části trajektorie. Posledním člen je jen u náběžné a sestupné hrany a udává stoupání nebo klesání v μm na jeden bod průběhu. Tento člen je zapsán na 7 desetinných míst. Znaménko před číslem udává, jestli jde o přírůstek nebo úbytek. Člen se vypočítá jako: [11]

$$H = \frac{A}{P_x}; \frac{20}{1282} = 0.0156006 \quad (6)$$

A je amplituda, P_x je délka dané části náběžné nebo sestupné hrany v počtu bodů. Výsledný průběh podle sledů příkazů je vidět na obr. 18.

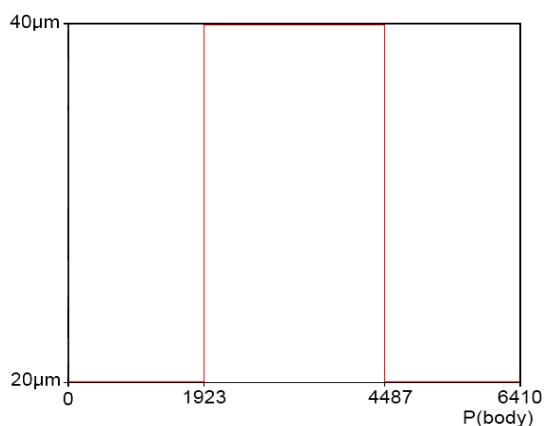


Obr. 18 výsledná trajektorie lichoběžníku

4.3.1.5 Tvorba příkazů pro Skokový a pilový průběh

Tvorba skokové a pilové trajektorie je v podstatě stejná jako lichoběžníková. Nejdříve si ukážeme tvorbu skokové trajektorie. Ze sledu příkazů je vidět, že jsou vynechány příkazy náběžných a sestupných hran trajektorie. Výsledný průběh je vidět na obr. 19 a zde je vidět daný sled příkazů. [11]

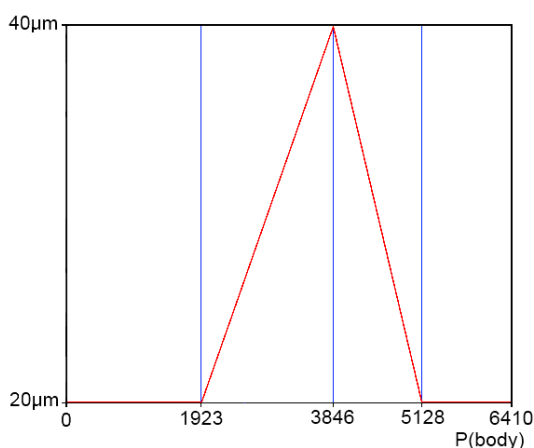
```
WAV A POL 0 1923 0 20.0000  
WAV A POL 1923 2564 0 40.0000  
WAV A POL 4487 1923 0 20.0000
```



Obr. 19 skoková trajektorie

Pilová trajektorie zase využívá náběžný a sestupných hran bez horní hrany trajektorie. Zde máme ukázkou příkazů a na obr. 20 je vidět výsledná trajektorie.[11]

```
WAV A POL 0 1923 0 20.0000  
WAV A POL 1923 1923 1923 20.0000  
WAV A POL 3846 1282 3846 40.0000  
WAV A POL 5128 1282 0 20.0000
```



Obr. 20 pilová trajektorie

4.3.1.6 Ostatní příkazy

Tyto příkazy jsou společné pro všechny druhy průběhů, určují počet opakování a další parametry pro daný průběh. Před zápisem samostatného průběhu musíme nejdříve vložit příkaz WGO A0, který zakazuje použití generátoru pro zadávaný průběh, což nám umožní zápis dané trajektorie. Dále následuje vlastní sled příkazů pro trajektorii a po něm následuje příkaz CFG, který je ve tvaru: [11]

WAV A CFG n m p k s L

n je počet bodů na periodu, m je konstanta pro přírůstek, p je fáze periody, k je hodnota přerušení před přírůstkem, s je rozdíl posunutí mezi jednotlivými pulsy, implicitně je nastavené na 0. L je limit amplitudy. Zde je příklad daného příkazu, poslední parametr L se neuvádí, když není použit: [11]

WAV A CFG 6410 1 0 3

Dalším důležitým příkazem je WGC, který udává počet opakování dané trajektorie
WGC A n

n je počet opakování trajektorie. Př.:

WGC A 10

Posledním příkazem je znovu WGO, ale tentokrát z příznakem A1, který povoluje generátor. Tento příkaz odstartuje celý průběh zadané trajektorie. Př.:

WGO A1 [11]

4.3.2 Inicializační příkazy

Zde máme vypsány jednotlivé inicializační příkazy, které nastaví jednotku E-516 do stavu online pro komunikaci přes sériový port a nastaví jednotku E-516 do základního nastavení. Jednotlivé příkazy jsou zde popsány a jsou posílány v daném sledu, jak jsou tu po sobě popsány.[11]

- *IDN? Se dotazuje na druh jednotky a verzi firmeware. [11]
- *ERR? Tento příkaz vrací kód chyby, jestli nastala. Když nenastane žádná chyba, příkaz vrátí kód chyby 0. [11]
- SAI? Vrací daný počet identifikátorů os, podle toho kolik jich je připojených. [11]
- ONL 1 Nastavuje zařízení do Online režimu. Příkaz ONL 0 nastavuje Offline režimu. Nevrací žádnou odpověď. [11]
- SVO A1 Nastavuje pro osu A servo-control režim. Parametr os jsou: A, B, C. V našem případě máme jen jednu osu a tou je osa A. Režim zapínáme příznakem 1 a vypínáme 0. [11]
- WGO A0 Zakazujeme ovládání časového průběhu signálu pro osu A. [11]
- MOV A0 Nastaví piezo-aktuátor na ose A na polohu 0. MOV nastavuje absolutně polohu. Hodnota se nastavuje v mikrometrech s přesností maximálně na jedno desetinné místo. Podmínkou je aby hodnota nepřesahovala rozmezí daného piezo-aktuátoru. [11]

4.3.3 Rámce pro zápis a čtení z jednotky E-516

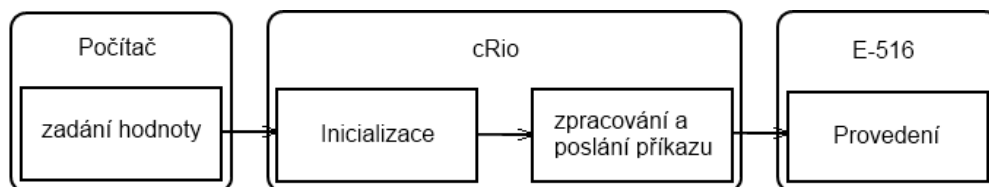
Komunikace je provedena UART protokolem, kde boud rate je nastaven na 57600. Data bits je 8, parita je none a stop bits je 1.0. [11]

Pro posílání jednotlivých příkazů jsme si vytvořili VI, které posílá příkazy v daných rámcích. Vstupem je pole pro dané příkazy, které se postupně pošlou do jednotky E-516. Další VI jsme si vytvořili pro čtení rámců z jednotky E-516. Čtení využívám jen, když je poslán příkaz, který vrací odpověď.

4.3.4 Logika programu pro řízení piezo-aktuátoru

4.3.4.1 Manuální řízení

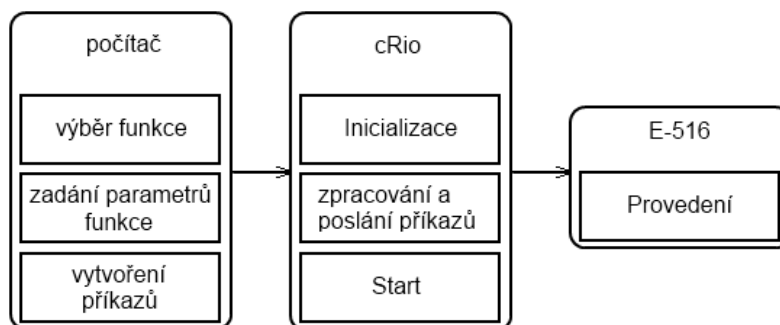
V manuálním módu nastavujeme jen posun piezo-aktuátoru. Blokové schéma funkce manuálního řízení máme na obr. 21. Kde v uživatelském prostředí zadáme požadovanou hodnotu polohy, z ní se nám vytvoří daný příkaz na provedení, který se po spuštění pošle do cRio. CRio nejdříve udělá inicializaci jednotky E-516 a následně na to se zpracuje poslaný příkaz a pošle se do jednotky E-516, která danou operaci provede.



Obr. 21 blokové schéma manuálního řízení

4.3.4.2 Programové řízení

V programovém řízení máme možnost nastavení trajektorie, které jsme si popsali v kapitole 4.3.1. Na obr. 22 je vidět blokové schéma programového řízení, které obsluhuje jednotlivé druhy trajektorii. V uživatelském prostředí na počítači si uživatel zvolí, o jaký druh trajektorie se má jednat (sinusová, lichoběžníková). Po výběru požadované funkce, zadá potřebné parametry pro vytvoření dané funkce, jsou-li parametry špatného rozsahu nebo chybného typu, uživatel je upozorněn chybovou hláškou na špatné zadání. Dále už uživatel zadává jen počet opakování dané trajektorie. Po zadání všech těchto náležitostí se vytvoří posloupnost příkazů, na základě zadaných parametrů. Dalším krokem je posílání všech příkazů do cRio. CRiu se nejdříve vykoná inicializace a po té se zpracují příkazy a postupně se nahrají do jednotky E-516, která čeká do doby, než je daná trajektorie spuštěna.

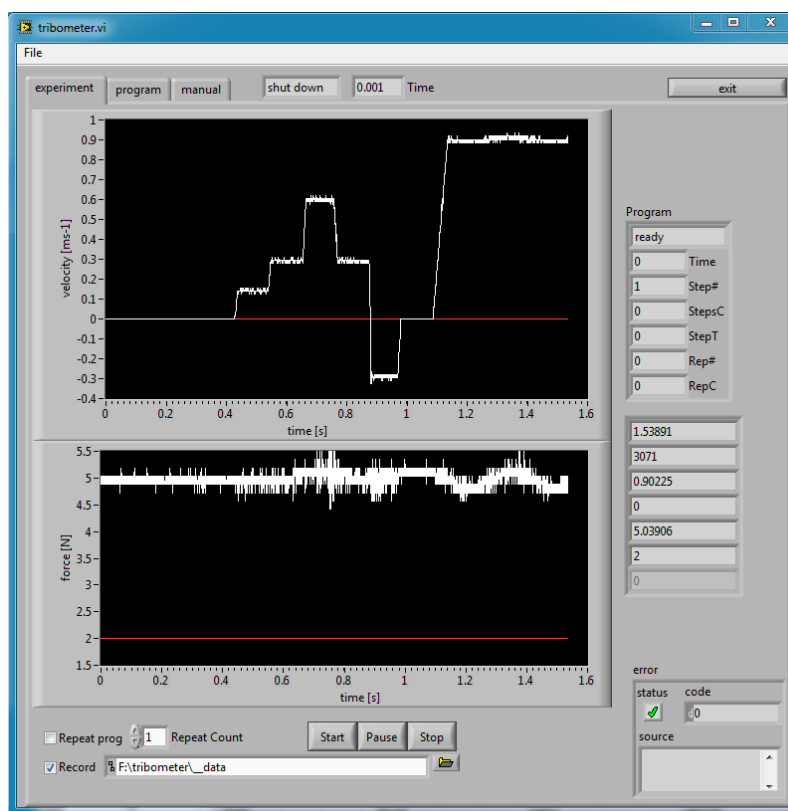


Obrázek 22 blokové schéma programového řízení

4.4 Realizace řídicího softwaru

4.4.1 Uživatelské rozhraní

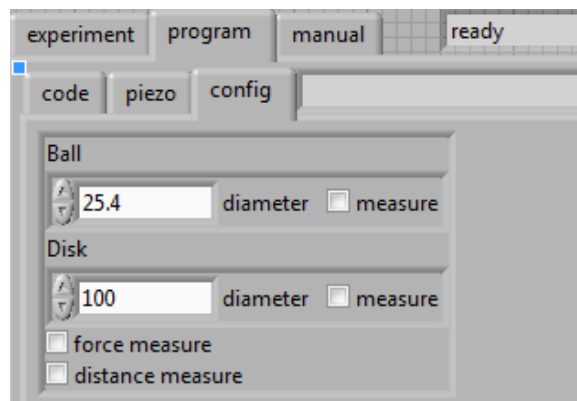
Na obr. 23 je vidět hlavní stránka uživatelského rozhraní. Vpravo nahoře je tlačítko exit, které nám ukončí celý program. V záložce file je nastavení snímačů pro měření. Dále jsou vidět tři záložky, první z nich je experiment, která je právě vidět na obr. 23. Další záložka je program, kde je nastavení programového řízení a piezo trajektorie. V poslední záložce je manuál, kde je manuální ovládání. V naší záložce experiment jsou vidět dva grafy. V horním grafu je vidět graf rychlosti kuličky. Graf rychlosti motoru prozatím není aktivní, protože není nainstalován enkodér na pohonu disku. V dolním grafu se zobrazuje síla z tenzometru. Pod grafem jsou tři tlačítka, první je start, které pouští nastavený program ze záložky program. Tlačítko pause je neaktivní. Tlačítko stop zastavuje probíhající program. Zaškrtnutá kolonka repeat program po zaškrtnutí umožňuje opakovat program podle nastavené hodnoty v kolonce repeat count. Zaškrtnutá kolonka record umožňuje uložení dat do textového souboru, který je pojmenován podle data a času měření. Cesta pro uložení se nastavuje před vlastním měřením. Vpravo dole je kolonka chybových hlášení (error), kde se vypisují jednotlivé chyby a jejich popis.



Obr. 23 uživatelské rozhraní

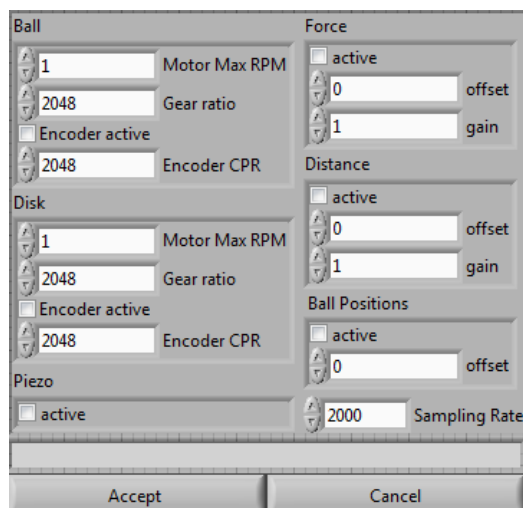
4.4.2 Nastavení

Máme vytvořené dvojce nastavení. V nastavení, které je v záložce program/config (obr 24) je zadání průměru kuličky a disku (diameter ball, disk), hodnoty jsou nastavitelné v milimetrech. Když nejsou nastavené žádné hodnoty, tak se nastaví výchozí nastavení. Je zde možnost měření průměrů, ale to zatím není realizováno. Měření síly a vzdálenosti (force, distance measure), je také předpřipravené pro budoucí úpravy programu. Nastavení průměrů je důležité, pro přepočítání obvodových rychlostí daných prvků.



Obr. 24 program/config

Další nastavení je v File/setting (obr. 25), kde se nastavuje u jednotlivých servomotorů maximální povolené otáčky (Motor Max RPM), pak přepočtení konstantu (Gear ratio), která vyjadřuje převod jednotlivých převodovek servomotorů. Další nastavení je zaškrtnutí, jestli je enkodér aktivní nebo vypnut (Encoder active). Další nastavení je CPR enkodéru (Encoder CPR), což vyjadřuje počet vzorků na otočení, implicitně je tato hodnota nastavená na 2048. Piezo nastavení je odškrtnutí kolonka, která určuje, jestli je piezo-aktuátor aktivní. V části menu síla (Force), se nastavuje parametry pro měření síly, první je odškrtnutí kolonka (active), která určuje, jestli je snímání síly aktivní, pod ním je nastavení offsetu (offset) a zisku (Gain). Další nastavení je měření vzdálenosti (Distance), kde je stejné nastavení jako u síly. Další nastavením je nastavení offsetu pro kuličku, offset je aktivní jeli zaškrtnuta kolonky (active). Poslední nastavení je počet vzorků za sekundu (Sampling Rate), nastavení tohoto parametru nám určuje počet snímaných vzorků, které budeme vyčítat z enkodérů a dalších snímačů. V našem případě je nastavená hodnota na 2000 vzorků za sekundu.

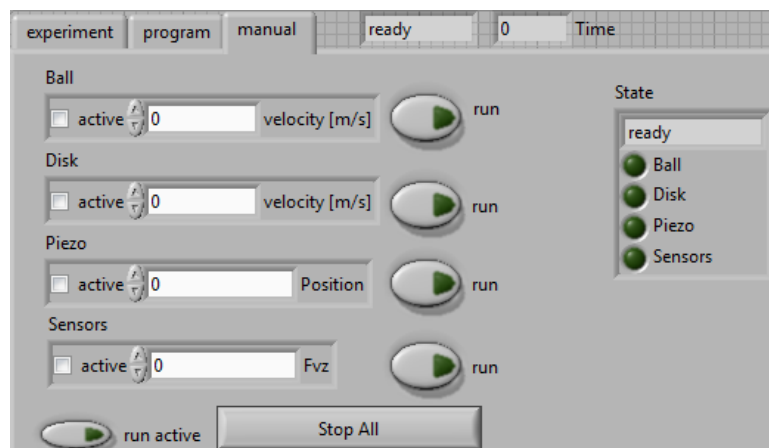


Obr. 25 file/setting

4.4.3 Manuální řízení

Manuální řízení (obr. 26) byl navržen pro jednoduché úkony. Máme několik možností výběru řízení v manuálním režimu. Do jednotlivých kolonek zadáváme příslušné hodnoty pro daný typ zařízení. První dvě nastavení jsou rychlosti pro servomotory, další hodnota na nastavení je zadání polohy piezo-aktuátoru. Nastavení Sensors se nevyplňuje,

prozatím je tato možnost nastavení neaktivní. Pomocí tlačítek run spouštíme jednotlivé části, například spuštěním servomotoru kuličky běží jen kulička zapnutím run u Pieza se kulička zastaví a nastaví se piezo-aktuátor. Jestliže chceme spustit zároveň několik položek v manuálním režimu, tak musíme použít zaškrtačací kolonky před požadovanými částmi nastavení a spustit tlačítkem run active. Ukazatel state nám zobrazuje, v jakém se nacházíme režimu a jaká zařízení jsou aktivní. Posledním tlačítkem je Stop All, které je nadřazeno všem akcím a po zmačknutí zastaví jakýkoliv vykonávaný proces.



Obr. 26 manuální ovládání

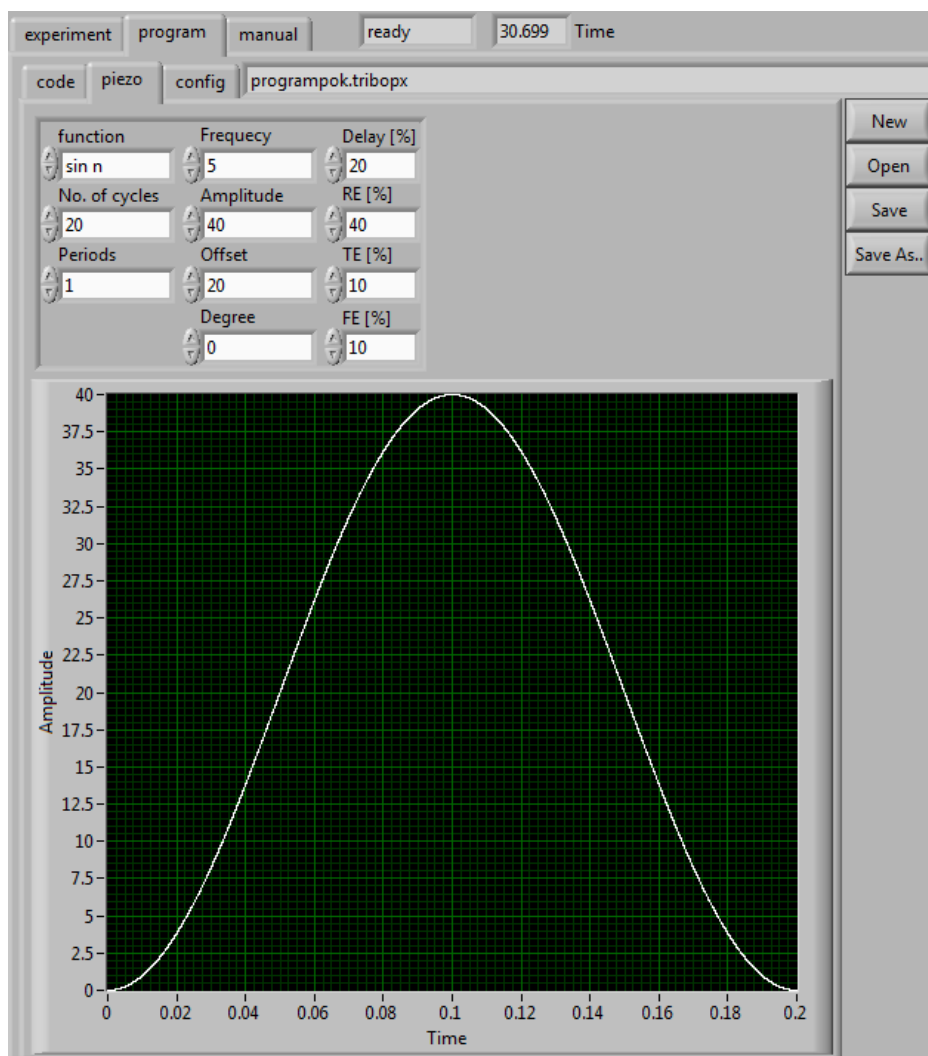
4.4.4 Nastavení trajektorie piezo-aktuátoru

Nastavení trajektorie je v záložce program/piezo. Na obr. 27 vidíme nastavení pro trajektorii piezo-aktuátoru. V kolonce function vybíráme ze sinusové (sin n) nebo lichoběžníkové trajektorie (trapezoid). Položka No. of cycles určuje počet opakování daného průběhu. Položka periods neovlivňuje průběh, jen nám umožňuje zobrazit více period v grafu. Na obrázku vidíme vybranou sinusovou trajektorii, která se nastaví pomocí několika parametrů. Logika vytváření vlastních příkazů trajektorie je popsán v kapitole 4.3.1.

V nastavení zadáme požadované hodnoty pro danou trajektorii, která se zobrazí v grafu a také se vytvoří příslušné příkazy pro danou trajektorii. Popis jednotlivých parametrů pro sinusovou trajektorii je v tab 6.

Položky	Vysvětlení
Frequency	Nastavení frekvence [Hz]
Amplitude	Nastavení amplitudy [μm]
Offset	Nastavení offsetu [μm]
Degree	Posunutí sinusové funkce [$^\circ$]

Tab. 6 sinusový průběh



Obr. 27 generátor trajektorií

Další je lichoběžníkový trajektorie, která se liší v nastavení oproti sinusové. Nastavení a vysvětlení parametrů je vidět v tab. 7. Kde kombinací jednotlivých parametrů můžeme vytvořit skokovou, lichoběžníkovou a pilovou trajektorii. Použitím všech parametrů vytváříme lichoběžníkový průběh, vynecháním náběžné a sestupné hrany vznikne skokový průběh. Využitím náběžné a sestupné hrany bez použití horní hrany, vytváříme pilový průběh. Hodnoty jsou zadávány v procentech, délka dané části se odvíjí od času periody.

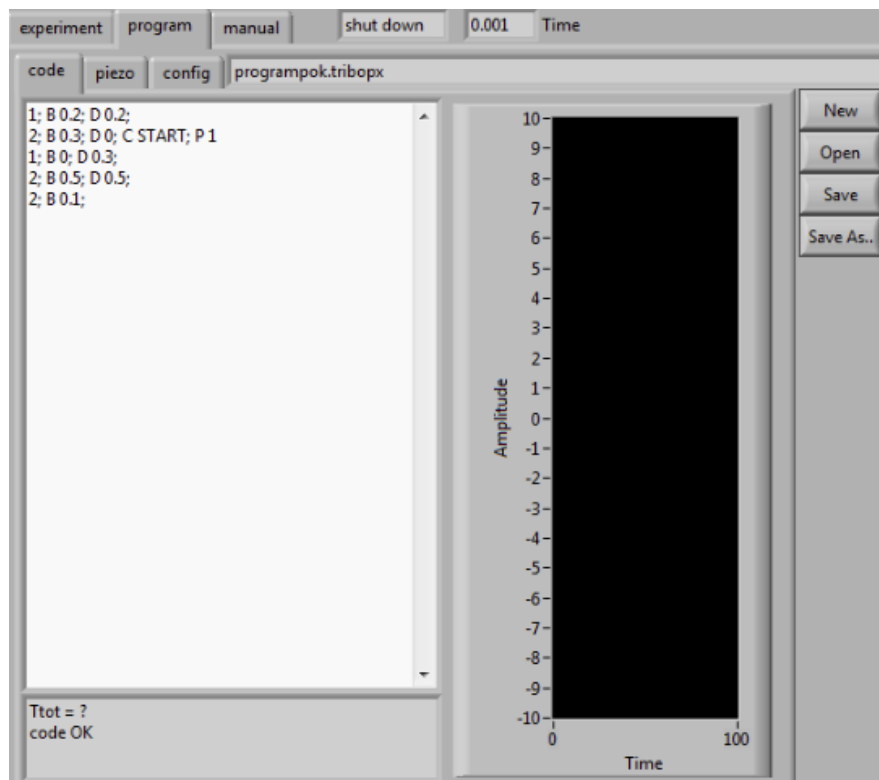
Položky	Vysvětlení
Delay	Úroveň zpoždění [%]
RE	Náběžná hrana [%]
TE	Horní hrana[%]
FE	Sestupná hrana[%]

Tab. 7 skokový průběh

4.4.5 Programové řízení

Programové řízení bylo navrženo jako textově zapisovaný program (obr. 28). V textovém poli nalevo je vidět zápis programu. Vpravo máme čtyři tlačítka. První tlačítko

New nám, vytvoří nový čistý program. Tlačítko open umožňuje otevřít už vytvořené programy. Další dvě tlačítka nám umožňují daný program uložit.



Obr. 28 programové rozhraní

Program má strukturu, že na prvním místě je vždy zadaný čas, a na dalších místech jsou jednotlivé příkazy pro vykonání. Jednotlivé instrukce jsou oddělovány středníkem. V tab. 8 jsou uvedeny všechny druhy povolených příkazů a jejich vysvětlení.

příkaz	formát	vysvětlení
	t;	t je časový usek programu, může nabývat hodnot od 0. [s]
B	B v;	B je příkaz pro zadání rychlosti kuličky, v je zadaná rychlost v [m/s], jeli rychlost záporná, servomotor se otáčí proti hodinovým ručičkám.
D	D v;	D je příkaz pro zadání rychlosti disku, v je zadaná rychlost v [m/s]], je-li rychlost záporná, servomotor se otáčí proti hodinovým ručičkám.
C	C START;	C je příkaz pro kameru kdy příznak START spouští kameru.
P	P 1;	P je příkaz pro piezo-aktuátor příznak 1 spouští přednastavený průběh.

Tab. 8 příkazy programu

Z našeho příkladu z obr. 28 je vidět že po uběhnutí jedné sekundy se spustí servomotor kuličky a disku. Jejich obvodová rychlost bude 0.2m/s, po uplynutí 2 sekund kulička zrychlí na 0.3m/s a disk se zastaví. Dále se spustí kamera a také přednastavený průběh piezo-aktuátoru, takhle se vykoná celý program.

Po vytvoření daného programu se překontroluje správnost zápisu. v první kontrole se zkontroluje, jestli jsou použity správné znaky pro dané příkazy a jestli nejsou použity další nepodporované příkazy. Také se zkontroluje pořadí jednotlivých příkazů, kde je podmínkou, že na každém řádku programu musí být na prvním místě čas. Další příkazy mohou mít různé pořadí, ale musejí mít správné parametry. Další kontrola se provádí u zadaných rychlostí, jestli nepřesahují povolenou hranici rychlosti z nastavení. Jestliže by

nastala nějaká ze zmíněných chyb, tak se vypíše chybová hláška s danou chybou. Nenastane-li žádná chyba, začne se daný program zpracovávat.

První krok při zpracovávání je, že se přidá nulový čas a nulové rychlosti pro servomotory. Dalším krokem je přepočet času na absolutní hodnoty a k nim jsou přiřazeny dané příkazy. Hodnoty rychlostí se přepočítají na ot/min. Po té se vytvoří body s přednastavenými rychlostmi a zrychleními, kde musíme zkontrolovat, jestli jsme nepřesáhli povolený počet míst, který je 8 pro rychlosti, s tím, že první místo vždy zabíráme pro 0 rychlost, takže ve skutečnosti máme jen 7 míst pro rychlosti. Dále máme 8 míst pro zrychlení a 8 míst pro zpomalení. Když přesáhneme povolený počet míst, bude nám vypisána chybová hláška. Ke každé zapsané rychlosti nebo zrychlení a zpomalení bude přiřazen index od 1 do 8. Jsou-li dvě nebo více rychlostí stejných, tak se zapíše jen jednou a k dalším je přiřazen jen daný index rychlosti, stejné je to i pro zrychlení a zpomalení. Takže se nám vytvoří dvě tabulky (obr. 29), v jedné budou zapsány jednotlivé body pro daný servomotor (spodní tabulka, obr. 29) a v další budou zapsány jednotlivé indexy (horní tabulka, obr. 29).

Time	B	D	C	P
1000	v 1 a 0	v 1 a 0	-1	-1
Time	B	D	C	P
3000	v 2 a 1	v 2 a 1	-1	-1
v 0	751.913	1127.87	v 0	763.944 1145.92
a	0.265988	0.132994	a	0.261798 0.174533 0
d	0.265988	0	d	0.1309 0

Obr. 29 tabulka bodů a indexů

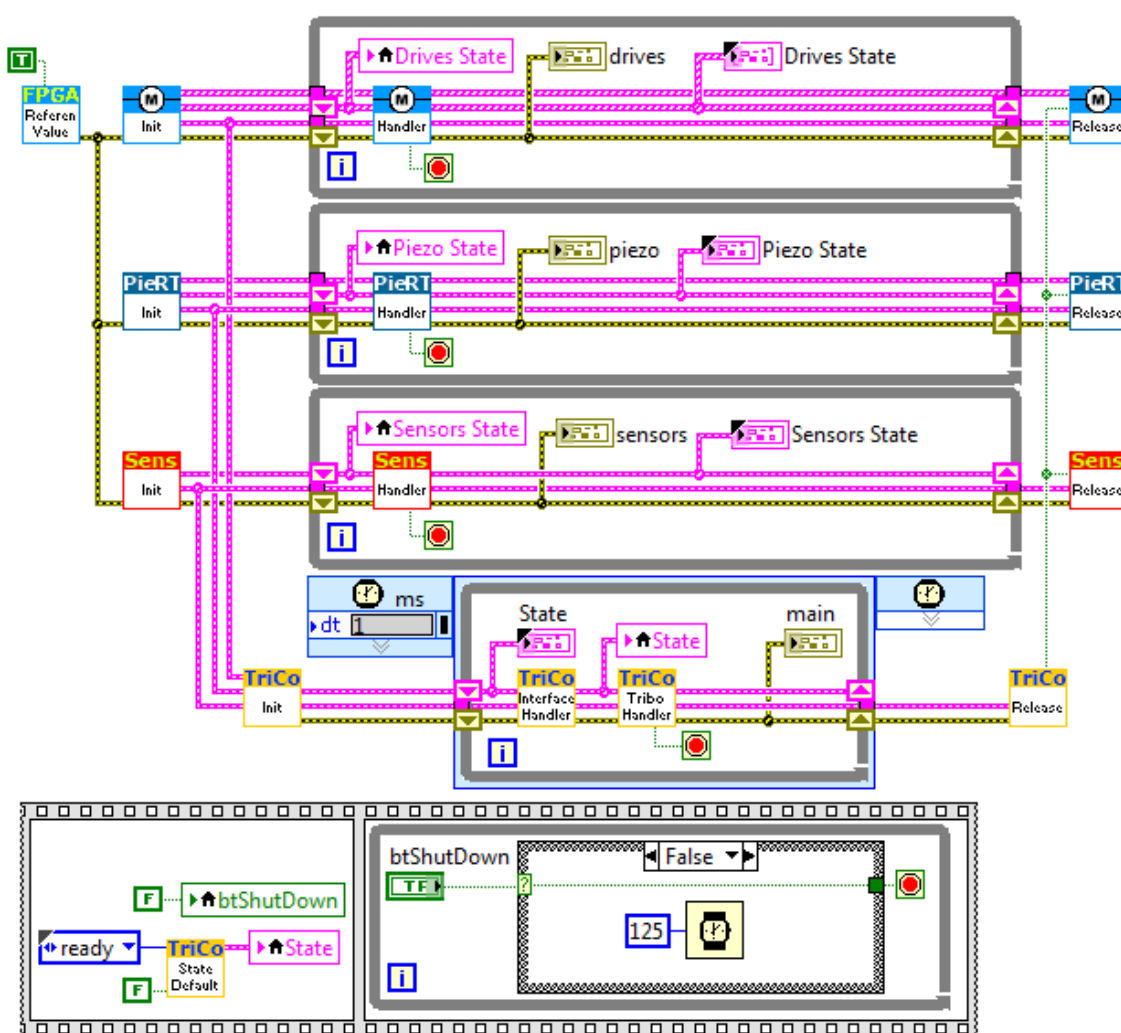
V dalším kroku je připojena tabulka s programem pro piezo-aktuátor, za předpokladu, jestli jsme piezo aktivovali v nastavení. Po vykonání všech těchto úkonů jsou tabulky programu poslány na cRio, kde se vykoná jednotlivá obsluha příkazů.

4.4.6 obsluha programu na cRio

Po poslání tabulek programu z uživatelského rozhraní na cRio se začne provádět obsluha programu v daném pořadí. Na obr. 30 je struktura programu na cRio. Po spuštění programu na cRio se nejdříve otevře reference na FPGA. V dalším kroku se nastaví komunikace pro jednotlivá zařízení. Na obrázku je vidíme jako subVI pojmenovaná Init. V časové smyčce se obsluhuje program, který je poslán z počítače a obsluhuje zbylé tři smyčky pro ovládání jednotlivých zařízení. První věc co se zjišťuje, po příchodu programu je o jaký režim se jedná, jestli jde o manuální nebo programové řízení. V dalším kroku je podle toho vybrán režim řízení. Vybraný režim se začne provádět a ovládat zbylé smyčky pro jednotlivá zařízení. V manuálním režimu se hned jednotlivé příkazy zapíše a vykonají, kdež to v programovém řízení se začnou zapisovat jednotlivé body do měničů a také program pro průběh piezo-aktuátoru. Čeká se do doby, než všechny příkazy budou zapsány do jednotlivých zařízení, což může trvat i několik sekund. Jestliže neproběhne

zápis nějakého příkazu je zápis opakován. Opakování je prováděno 3x, jestliže i po třetím průchodu se data nezapiší, pokračuje se dále a daný krok se nevykoná. Po zápisu všech dat se čáka na start, ten odstartuje celý experiment.

V programovém režimu se začnou vykonávat jednotlivé řádky programu, které vidíme v obr. 25 v horní tabulce. Obsluha programu je provedena na ve smyčce, kde se kontroluje čas běhu programu s jednotlivými časy kroků poslaného programu, když je překročen čas programu proti nastavené době přechází se na další krok a vykonávají se dané příkazy pro jednotlivé výstupy z cRia do jednotlivých zařízení. Jestliže je i puštěné měření, tak se začnou vyčítat data z jednotlivých snímačů a ukládat do fronty, která je postupně vyčítána a postupně posílána do uživatelského rozhraní v počítači na uložení a zobrazení výsledků experimentu. Program se vykonává do doby, kdy jsou všechny příkazy zpracovány a všechny zařízení a snímače vrátili odpovědi. Program na cRiu se přepne do vyčkávání a čeká na další úkony.



Obr. 30 Struktura programu na cRiu

5 Závěr

Cílem bakalářské práce bylo navrhnout a realizovat software pro řízení tribologické soustavy. Po částech jsme se seznámili s danou problematikou, a postupně jsme se seznámily s danými zařízeními.

Základem bylo navržení daných rámců pro komunikaci s jednotlivými zařízeními. Nejdříve jsme navrhli přímé ovládání mezi jednotlivými zařízeními a uživatelským rozhraním na počítači, které jsme použili pro seznámení s danými zařízeními. Po odzkoušení se daný software upravil pro CompactRio. Kde se naprogramovala jednotlivá obsluha zařízení pomocí zásuvných modulů. Rámce na posílání jednotlivých příkazů pro daná zařízení. Realizované bylo také zpracování dat ze senzorů.

Dále jsme realizovali celkový software pro tribometr, ohled byl brán na budoucí variabilitu. Uživatelské rozhraní běží na počítači, v něm je možné volit různá nastavení pro zařízení a také konfigurace snímačů pro měření a jejich parametry. Uživatel má možnost z dvou hlavních druhů řízení, manuálního a programového. Manuální řízení je jen pro jednoduché řízení experimentu. V programovém řízení máme možnost nastavení trajektorii pro piezo-aktuátor. V současnosti máme výběr ze sinusové a lichoběžníkové trajektorie.

Vlastní program řízení je psán textově, podle námi zavedených příkazu. Tvorba samostatného programu je popsána v bakalářské práci. Omezení programu jsou dány použitými zařízeními a jejich vlastnostmi.

Dále je realizováno měření daného experimentu. Realizované máme měření z enkodérů, přičemž prozatím máme nainstalován jen jeden enkodér, který je nainstalován na pohonu kuličky. Dále máme realizováno měření síly a vzdálenosti. Výsledky měření se nám zobrazují v uživatelském rozhraní softwaru, kde máme možnost uložení dat do textového souboru.

Daná práce bude dále rozvíjena o další možnosti měření a tvorbu piezo trajektorií. Případně o další požadavky ze strany Ústavu konstruování na daný experiment.

6 Seznam použité literatury

- [1] TU V LIBERCI. 5. *Tribologie* [skripta]. [cit. 2012-05-23]. Dostupné z:
http://www.ksp.tul.cz/cz/kpt/obsah/vyuka/stud_materialy/ttv/tribologie.pdf
- [2] NI cRIO-9024. NATIONAL INSTRUMENT. *www.NI.com* [online]. [cit. 21.5.2012].
Dostupné z: <http://sine.ni.com/ds/app/doc/p/id/ds-199/lang/en>
- [3] NI 9870. NATIONAL INSTRUMENT. *www.NI.com* [online]. [cit. 21.5.2012].
Dostupné z: <http://sine.ni.com/nips/cds/print/p/lang/en/nid/204259>
- [4] NI 9239. NATIONAL INSTRUMENT. *www.NI.com* [online]. [cit. 21.5.2012].
Dostupné z: <http://sine.ni.com/ds/app/doc/p/id/ds-199/lang/en>
- [5] NI 9237. NATIONAL INSTRUMENT. *www.NI.com* [online]. [cit. 2012-05-21].
Dostupné z: <http://sine.ni.com/nips/cds/print/p/lang/en/nid/208791>
- [6] NI 9411. NATIONAL INSTRUMENT. *www.NI.com* [online]. [cit. 2012-05-21].
Dostupné z: <http://sine.ni.com/nips/cds/view/p/lang/en/nid/208812>
- [7] CONTROL TECHNIQUES. *SP rozšířený návod* [Datasheet]. 9.9.2009. [cit. 2012-05-23]. Dostupné z:
http://www.controltechniques.cz/PDF/SP_Rozsireny_navod_1.pdf
- [8] VUES. *M254 servomotor* [Datasheet]. [cit. 2012-05-23]. Dostupné z:
http://www.vues.cz/doc/CZ_M25_070927.PDF?docid=112
- [9] UNIDRIVE SP. EMERSON. [online]. [cit. 2012-05-23]. Dostupné z:
http://www.emersonindustrial.com/ImagesControlTechniques/PR/PressReleases/StemaService/unidrive_sp.jpg
- [10] AC Servo Motor Bautz M256D-O-NF40. [online]. [cit. 2012-05-23]. Dostupné z:
<http://www.technic.co.uk/UserFiles/images/5809070709LZ0085.JPG>
- [11] PHYSIC INSTRUMENT. *PZ 102E User manuál* [Datasheet]. 2005-09-23. [cit. 2012-05-23].
- [12] PHYSIC INSTRUMENT. *PY 112E Software manual: Wave Editor for E-516 Wave* [Datasheet]. 2005-03-14. [cit. 2012-05-23].